Preprint version; Final version at https://ieeexplore.ieee.org/document/10803021
IEEE Robotics and Automation Letters (RA-L)
DOI: 10.1109/LRA.2024.3518104

Garment Diffusion Models for Robot-Assisted Dressing

Stelios Kotsovolis[1] and Yiannis Demiris[1]

***Abstract*—Robots have the potential to assist people with disabilities and the elderly. One of the most common and burdensome tasks for caregivers is dressing. Two challenges of robot-assisted dressing are modeling the dynamics of garments and handling visual occlusions that obstruct the perception of the full state of the garment due to the proximity between the garment, the robot, and the human. In this paper, we propose a diffusion-based dynamics model for garments during robot-assisted dressing that can deal with partial point cloud observations. The diffusion model, conditioned on the observation and the robot's action, is used to predict a full point cloud of the garment's opening of the future state. The model is utilized in a model predictive controller, that is trained iteratively with model-based reinforcement learning. In our experiments, we examine a common problem of dressing: the insertion of a garment's sleeve on an arm. As demonstrated by the performed experiments, the proposed diffusion-based model predictive controller can be effectively used for robot-assisted dressing and handle visual occlusions. Moreover, our approach is highly sample-efficient. Specifically, the controller achieved 91.2% success rate in the examined dressing task with less than 100 sampled trajectories. Real-wold experiments demonstrate that the proposed method can adapt to the sim-to-real gap and generalize well to novel garments and configurations of the body.**

***Index Terms*—Physical Human-Robot Interaction, Model Learning for Control, Human-Centered Robotics, Robot-Assisted Dressing**



Fig. 1. Garment diffusion models can efficiently learn the dynamics of garments from partial observations in a robot-assisted dressing task, enabling a bi-manual Baxter humanoid robot to dress a medical-training manikin's arm.

I. INTRODUCTION

DRESSING is one of the most frequent activities of daily living that people require assistance with. At the same time, it is one of the most difficult activities for caregivers to perform [1]. Robots could potentially assist with dressing for elderly people or people who suffer from impairments. Assistive dressing remains a very challenging task for robots, because of the difficult task of garment manipulation, the visual occlusions of the garments and the human body and safety. Specifically, garments, being deformable objects with infinite degrees of freedom, present challenges for robots in representing their state and modeling their dynamics.

Early studies mainly used either the human body's pose [2] or force feedback [3] in the controller's architecture, while image observations were usually used before the dressing task, mainly for grasping[4]. Raw visual observations as feedback in the dressing process were utilized more recently. In [5], point cloud observations were used as input in a reinforcement learning policy. In our latest work [6], we used the point cloud of the garment's opening to construct a graph-based state representation, which was processed by a graph neural network to provide dynamic predictions for future states. However, a limitation of our previous work was the assumption of a fully observable opening of the garment.

In this paper, we lift this assumption and study garment dynamics modeling under partial visual observations to predict their behavior for control in robot-assisted dressing. We note at this point, that this paper aims to propose and examine methods for capitalizing on visual information and dynamics during a robot-assisted dressing task and not to show superiority of vision against other modalities, such as force. Our approach can be potentially complemented by force models with approaches such as [7], but this is not examined in this study.

In particular, we propose a dynamics model that utilizes partial observations to predict the future point cloud of the garment's opening. Driven by recent advancements in generative models and reconstruction techniques from partial information by diffusion models, we propose a diffusion-based dynamics model for garments from partial point cloud observations. Specifically, we use the partial point clouds of the garment as well as a reconstructed point cloud of the body, along with the action of the robot to condition a diffusion model, which generates the full point cloud of the opening for the next time-step of the dressing process. The model is utilized in a model predictive controller to select optimal actions. To boost the performance and train the model with on-policy samples, we alternate between training and sampling, as in model-based reinforcement learning [8].

[1]The authors are with the Personal Robotics Lab, Department of Electrical and Electronic Engineering, Imperial College London, London SW7 2BT, U.K. (e-mail:s.kotsovolis21@imperial.ac.uk; y.demiris@imperial.ac.uk).

This work was supported in part by UKRI Grant EP/V026682/1, and a Royal Academy of Engineering Chair in Emerging Technologies to YD. For the purpose of open access, the author(s) has applied a Creative Commons Attribution (CC BY) license to any Accepted Manuscript version arising.

We focus on a common task of robot-assisted dressing: the insertion of a garment's sleeve around a human's arm. To safely train the model, a simulated environment of the examined dressing task was constructed. We demonstrate in our experiments that our model can be transferred in a zero-shot manner in real-world dressing scenarios. To evaluate the efficiency of our approach, a comparison with several state-of-the-art approaches was conducted. Overall, the contributions of this paper can be summarized as follows:

- A model-based approach for robot-assisted dressing from partial point cloud observations
- A diffusion-based model for garment dynamics
- A comprehensive evaluation in simulation and real-world dressing scenarios with a Baxter humanoid robot and a medical-training manikin, as shown in Fig. 1.

II. RELATED WORK

A. Robot-assisted Dressing

Several studies have been conducted regarding robot-assisted dressing to tackle different challenges of the task. One category assumes collaborative users with the ability to move during the task. These studies usually focus on adaptation to the user's motion under the occlusions of the garments [9], [10], [11] of the garment and/or providing personalized assistance to the users [12], [13], [14]. Another category focuses on the problem of garment manipulation in preparation for the dressing task. Grasping and unfolding garments are the main objectives of such studies aiming to bring garments in a suitable dressing state [15], [16], [17].

Another line of work focuses, as we do, on the garments' manipulation during the task and proposes control methods to bring them to a desired state around the human body. Some studies proposed imitation and reinforcement learning [18], [19], [20] to control the garments. However, the motion of the robot is based on the configuration of the human body and/or sparse fully observable key points of the garments. Some works propose force feedback controllers [21], [2] or predictive controllers with force models [3], [22]. Control with raw visual observations as feedback was recently approached in [5], where a model-free reinforcement approach was proposed and combined later with force models [7]. From the aforementioned, we can conclude that most studies use reinforcement or imitation learning to train policies, while model-based approaches are used mainly with force models. However, garment models with visual feedback can also be beneficial for dressing, as model-based policies tend to be more sample efficient and offer better generalization [23], [24]. Although we proposed a learned graph model for dressing in our previous work [6], the approach suffered from the assumption of partial observations. In this work, we propose garment dynamics models for assistive dressing based on partial visual observations.

B. Deformable Object Manipulation

Deformable object manipulation is still a challenging problem for robots, because of the difficulties in representing their state space and modeling their dynamics. Different methods have been proposed for problems, such as garment folding [25], unfolding [26], and smoothing [27]. Some studies focus, as we do, on dynamics modeling for deformable objects, that are used in predictive controllers for control [28], [29]. A few works have addressed the challenge of partial observability [30], [31]. However, two important differences with our study are that the usual setup is two-dimensional with the deformable object placed on a table, while pick-and-place actions are used for the studied tasks. In this work, we focus on robot-assisted dressing, a 3D problem with a highly contact-rich environment caused by garment-body interactions. Moreover, closed-loop actions are needed to bring the garment to a desired state around the body. For this reason, we propose a model predictive controller with a garment dynamics model that uses point clouds both as input and output to predict the garment's behavior.

C. Diffusion Models in Robotics

A few works have explored diffusion models in different areas of robotics. In [32] diffusion models were used to learn cost functions for selecting optimal actions, while in [33], [34] for action generation from demonstrations. In model-based approaches, diffusion has been used for world modeling in Dyna-based MBRL frameworks [35], [36], as well as in MPC-based frameworks [37], [38]. Following these studies, we use a diffusion model in an MPC-based approach. However, we focus on the case of partial observations in a 3D contact-rich deformable object manipulation environment. Therefore, we use point clouds as the environment's representation and we specifically model the garment's dynamics by conditioning the diffusion process on the partially observed point cloud and the robot's action. In the context of deformable object manipulation, diffusion models have been used only in a few studies either for imitation learning [39], [40] or to compose skill trajectories [41]. However, to the best of our knowledge diffusion models have not been used previously as a dynamics model for deformable objects.

III. DRESSING WITH GARMENT DIFFUSION MODELS

A. Problem Formulation and Assumptions

In this paper, we study the following problem of robot-assisted dressing. A bi-manual robot grasps a garment around its sleeve's opening on two opposing sides. We use bi-manual manipulation to have more control points over the garment to avoid failure scenarios of the garment getting caught on the arm, which occur especially often during the initial insertion of the garment around the palm. The goal of the task is to dress the garment's sleeve around the user's arm up to the shoulder. In this study, we make two assumptions. Firstly, since pose and shape estimation under occlusions caused by the garment is not the aim of this study, the user is assumed to remain still and we obtain the shape of the arm before the start of the dressing process. This assumption has been used in prior works that focus on the control aspect of the assisted dressing [5], [6], [3]. Secondly, we assume the garment is grasped at the beginning of the task. Our approach can be complemented

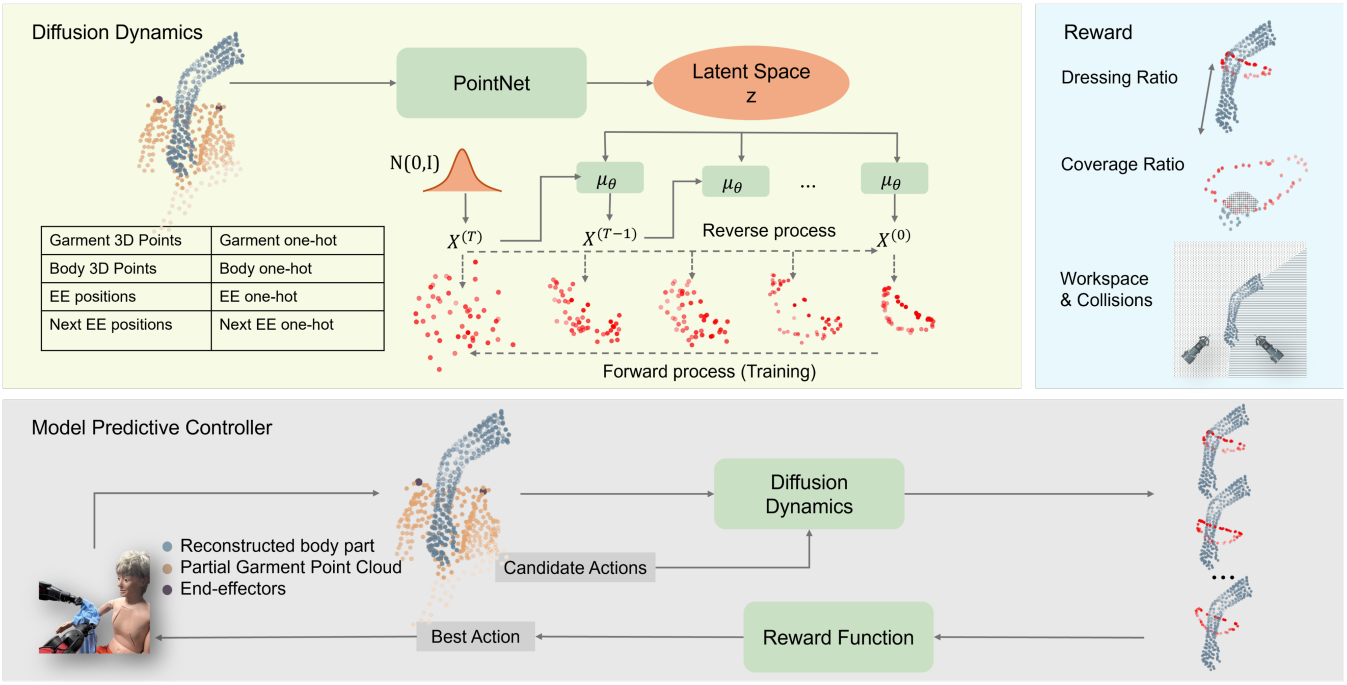


Fig. 2. Architectural diagram of the proposed diffusion-based MPC framework. **Top left:** The core module of our method is a garment diffusion dynamics model from point cloud observations. The point clouds of the garment and the arm, the robot's state and the action are represented in a single point cloud. The point cloud is encoded in a latent space z through a PointNet network. The latent space is used to condition a diffusion model. The diffusion model iteratively denoises an initial point cloud drawn from a normal distribution and outputs the predicted point cloud of the garment's opening. **Bottom:** In every time step, the MPC samples actions and a prediction is made for every one of them. The controller then selects the action that maximizes a reward function. **Top right:** The designed reward function is constructed by 3 terms. Firstly, it maximizes the dressing ratio. Secondly, during the insertion stage, it maximizes the coverage area of the opening around the hand. Finally, it defines the workspace of the two end-effectors and avoids collisions.

by grasping approaches such as [17], [15], but this is not the focus of this paper.

We formulate the dressing control task as a partially observable Markov decision process (POMDP) and use point clouds as observations. At each time step t, the robot being in a hidden state of the environment s_t, receives a point cloud observation o_t, takes an action a_t, moves to the state s_{t+1} and receives a reward R_{t+1}. We define the robot action for our task as a 6-dimensional vector, composed of the two end-effectors' positions in the Cartesian space. We use the remaining degrees of freedom to avoid collisions, as we describe later in section III-F. We solve the POMDP, with model predictive control (MPC), trained iteratively with model-based reinforcement learning (MBRL). An architectural diagram of our approach is depicted in Fig. 2. In this section, we discuss our approach in detail.

B. Diffusion Dynamics

The core of our method is a dynamics model that receives as input the current observation o_t and a robot's action a_t. The observation is composed of the garment's partial point cloud, a reconstructed point cloud of the user's arm, and the robot's end-effectors' positions. The output of our model is the full point cloud of the next time step's opening of the sleeve, which we mark as $\tilde{o}_{t+1}$. Therefore, the dynamics model can be described as a function f_d that predicts the future point cloud of interest, based on the current observation and the robot's action:

$$\tilde{o}_{t+1} = f_d(o_t, a_t) \tag{1}$$

The dynamics problem has been reduced to a point cloud to point cloud translation problem. We propose a diffusion model as our function f_d. Diffusion models have shown promising results in reconstruction tasks and recovery of missing information [42]. We base our formulation on [43], where point cloud diffusion models were proposed as point cloud generators and auto-encoders. In our case, we do not reconstruct the opening, but we predict the next step's configuration of the opening. To encode the action and capture the relationship between the different kinds of point clouds, we combine the observation-action pair in one point cloud as follows. The garments' partial point cloud, the body's point cloud, the 3D end-effectors' positions, and the robot's action are concatenated in a single point cloud. For each of the four different types of points in the point cloud, a one-hot vector is added in the feature vector, similar to [5]. The resulting point cloud is then encoded into a latent space z through a PointNet neural network [44].

We then use the latent encoding z to condition a diffusion model that outputs the next step's point cloud of the garment's opening. To avoid overloading the notation we define the predicted point cloud of the opening $\tilde{o}_{t+1}$ as $X^{(0)} = \{x_i^{(0)}\}_{i=1}^{N}$,

where N the number of points. We assume that the point cloud $X^{(0)}$ can be sampled independently from a distribution $q(x_i^{(0)}|z)$. A forward diffusion process aims to add noise to the point cloud in each step of the diffusion process t', and can be modeled as a Markov chain:

$$q(x_i^{(1:T)}|x_i^{(0)}) = \prod_{t'=1}^{T} q(x_i^{(t')}|x_i^{(t'-1)}) \tag{2}$$

where T are the steps of the process and

$$q(x^{(t')}|x^{(t'-1)}) = \mathcal{N}(x^{(t')}|\sqrt{(1-\beta_{t'})}x^{(t'-1)}, \beta_{t'}I) \tag{3}$$

is the diffusion kernel [45]. On the other hand, the reverse diffusion process aims to remove noise in each step of the process t' to generate the final predicted point cloud:

$$p_\theta(x^{(0:T)}|z) = p(x^{(T)}) \prod_{t'=1}^{T} p_\theta(x^{(t'-1)}|x^{(t')}, z) \tag{4}$$

$$p_\theta(x^{(t'-1)}|x^{(t')}, z) = \mathcal{N}(x^{(t'-1)}|\mu_\theta(x^{(t')}, t', z), \beta_{t'}I) \tag{5}$$

where β_t are hyper-parameters of the rate of the process, μ_θ is a point-wise neural network, and $p(x^{(T)})$ is the initial noise distribution, set to a normal distribution $\mathcal{N}(0, I)$.

The forward process is only used in the training process, as its functionality is to add noise to generate samples for training the network μ_θ of the reverse process. At inference, only the reverse process is used. The reverse process, conditioned on the latent space z, which represents the observation and action of the robot, iteratively denoises the initial point cloud $x^{(T)}$ of the process to generate the predicted point cloud of the opening of the next step $\tilde{o}_{t+1} = X^{(0)}$.

C. Model predictive control

We use the proposed diffusion-based dynamics model in a sampling-based MPC to select optimal robot actions. The controller samples a number of candidate actions a_s. Since the actions are defined as end-effectors positions, steps from a uniform distribution $[-\delta, \delta]$ are used to construct them. δ here is the maximum displacement allowed during an action. For each of the candidate actions, a point cloud of the next step's opening is predicted by the diffusion dynamics. Then, the action that leads to the highest reward is selected :

$$a_t = \underset{a_s}{\operatorname{argmax}}(\tilde{R}_{t+1}^{a_s}) \tag{6}$$

where $\tilde{R}_{t+1}^{a_s}$ is the expected reward under the action a_s.The reward of the examined dressing task is composed by 3 terms as follows:

$$R = R_{prog} + w * R_{cov} + R_a \tag{7}$$

where w is a tunable weight. To explain the 3 terms, we define two stages: before and after the initial insertion of the garment around the palm. We classify the stages based on whether the predicted opening is around the arm. The first term aims to reward the progress of the dressing task, similar to [5]. Firstly, we calculate the intersection point of the predicted opening with the arm. Then, R_{prog} is set as the ratio of the arm dressed up to the intersection point to the length of the whole arm. If the opening is not around the arm (first stage) we set $R_{prog} = -d_{arm}$, where d_{arm} is the distance between the center of the predicted opening and the hand. The second term R_{cov} is active only during the first stage (insertion) and aims to maximize the coverage of the opening around the hand. We set R_{cov} as the coverage ratio, calculated as a ratio of the number of hand's points that are inside the polygon of the opening, to the number of points of the full hand's point cloud. Finally, the third term R_a enforces penalties to prevent undesired behaviors. We define two half-spaces, which represent the preferred operation spaces of the two end-effectors, on two sides of the arm and we penalize the actions for which the end-effectors cross to the other half-space. We also penalize actions for which the end-effectors move away from each other further than a set distance. Finally, we introduce penalties when the end-effectors come closer than a set distance to the body's points with each other to avoid collisions.

D. Training

We trained our dynamics model entirely in simulation. Four front-open shirts with different sleeve lengths were constructed in Blender, and PyBullet was used as our physics simulator. Human arm meshes with varying poses were generated using the SMPL-X human model [46]. We constructed our environment based on the Assistive Gym simulation suite [47]. The two end-effectors were simulated by two anchor points on two opposing sites of the garment. During training, we applied domain randomization for the physics parameters, the size, and the initial position of the garment. We also randomized the body size and the configuration of the arm by randomly selecting poses. We used poses from the following ranges: [-30,30] for the lifting angle of the shoulder, [-30,30] for the lifting angle of the elbow, and [0,60] for the inwards bending of the elbow. A virtual camera was used to receive the partial point cloud of the garment, and its position was also randomized. Moreover, two generic grippers were used to simulate visual occlusions created on the point cloud around the grasping area.

A typical way of gathering data for a dynamics model is to perform random actions of the model and record the states before and after the actions. However, that way we can not ensure that our model has experienced transitions that occur during the dressing task. For instance, when performing random actions to collect data, it is uncommon to reach states where the garment is around the upper arm or even the forearm. To solve this problem, we alternated between training the dynamics model and sampling trajectories by deploying the MPC controller, as in model-based reinforcement learning. This way, we collect on-policy samples during the task, and the dynamics model experiences transitions throughout the whole task. In the first iteration, the data were collected by performing random actions. Moreover, we applied Gaussian noise in the point cloud observations to make our model

more robust to real-world noisy observations. We trained the diffusion model with the diffusion loss as in [43].

E. Collision Avoidance

While the controller handles the collisions of the end-effectors with the human body, collisions between the robotic arms and the human body should also be avoided to ensure safety. We noticed that such collisions occur often in such a dense workspace. For that reason, in real-world dressing scenarios, we deploy the MPC as a high-priority action and design a lower-priority action to avoid collision. Since the proposed MPC controls the cartesian position of the robot, the task's dimensions for each end-effector are 3D. We use the redundant degrees of freedom for collision avoidance. Let R be a set of 3D points of a robotic arm and B a set of points of the body. Firstly, we calculate the critical points for the two sets $C_r \in R$ and $C_b \in B$ for which the distance is minimum across the points of the two sets. We then, use a secondary action in the null space to take advantage of the redundant degrees of freedom and drive C_r away from C_b. We design our controller based on the obstacle avoidance formulation similar to [48]:

$$\dot{q} = J_p^{\dagger} K_1 (P_d - P) + (I - J_p^{\dagger} J_p) J_{C_r}^{\dagger} \dot{x_{C_r}} \tag{8}$$

where $\dot{q}$ is the joint velocity, $J_p^{\dagger}$ is the pseudo-inverse of the robot's linear velocity's Jacobian matrix J_p. P_d is the desired position of the end-effector, as selected by the model predictive controller, P is the current position of the end-effectors and K_1 is a tunable gain. J_{C_r} is the linear velocity jacobian of the robot's critical point and $J_{C_r}^{\dagger}$ is its pseudo-inverse. Finally, $\dot{x_{C_r}}$ is an escape velocity that drives C_r away from C_b.

The first term represents the primary action and is a cartesian position controller, which executes the MPC's action. The second term is activated only when the distance between the critical points is smaller than a threshold and aims to drive the robot's critical point away from the body's critical point in order to avoid collisions.

F. Shape Estimation

As mentioned above, we reconstruct the shape of the arm dressed and we use its full point cloud. Although this is not the aim of this paper, we briefly discuss the reconstruction method for the sake of completion. We use PyMAF-X [49] a mesh reconstruction method from RGB images to generate the shape parameters of the SMPL-X model. We observed that using the same method for predicting the 3D joints did not result in the desired accuracy for the task, especially because of errors in the depth of the prediction. Instead, we predicted the 2D joint positions by using MediaPipe [50] and then used the depth image to acquire the 3D joint positions. Finally, by using the estimated shape and joint parameters we generated an accurate SMPL-X mesh of the arm.

IV. Experiments and Results

To investigate the performance and efficiency of the proposed dressing system, we conduct experiments both in simulation and in the real world.

A. Simulation

We compare the proposed diffusion-based model predictive controller (Diff-MPC) with the following baselines and ablations in simulation:

- **Diff-MPC f.e.**: To investigate the importance of action encoding, we replace the one-hot vector action encoding in the proposed dynamics model with a feature encoding (f.e.) of the end-effectors' action.
- **Diff-MPC w/o MBRL**: To investigate the importance of sampling on-policy data with model-based reinforcement learning, we drop the iterative sampling during training. Instead, we initially gather all data by performing random actions and then train the dynamics model.
- **PC-GAN MPC**: To investigate the effectiveness of the diffusion-based dynamics we substitute our diffusion model with PC-GAN [51], a state-of-the-art generative model for point clouds.
- **VAE MPC**: Similarly, in this baseline, we substitute the diffusion model with a Variational Autoencoder with a PointNet encoder, similar you ours.
- **TD-MPC [52]**: To investigate the effectiveness of the proposed diffusion-based controller we compare it against TD-MPC, a state-of-the-art MBRL algorithm. We use the authors' implementation, with a PointNet encoder similar to ours.
- **SAC [53]**: We compare against Soft Actor-Critic, a state-of-the-art model-free RL algorithm. We use Ray Rllib's [54] implementation with a PointNet encoder similar to ours.
- **SAC-distilled [5]**: We use SAC with policy distillation which was proposed to improve the performance of SAC for robot-assisted dressing. We trained 27 teacher policies, one for each of 27 subranges of the arm's configurations, used to guide a final distilled policy. We note, that although the approach was proposed for dressing, there are several differences in the examined experiments (bi-manual manipulation, observation and action space, experimental and simulation setup), making direct comparisons not always possible.

We investigate the performance of our method and the baselines by measuring the achieved dressing ratio of the arm, the results of which are shown in Fig. 3. To investigate how efficient our method is regarding the sampled trajectories needed to train, we compare against the last 3 baselines, which are different ways of iterative training. The learning curves are shown in Fig. 4.

The following conclusions can be made from the results of the simulation experiments. The action encoding does not have a significant role in the performance of our method, as the two encoding methods result in a very similar performance, with the one-hot encoding method being slightly better. We also observe that when iterative sampling and training with MBRL is not used, the MPC performs poorly. This can be explained by the fact that the dynamics model is trained with interactions close to the initial state, but is unfamiliar with the dynamics of the late stages of the dressing task. Therefore, it can only dress the arm up to a certain point where the dynamics predictions

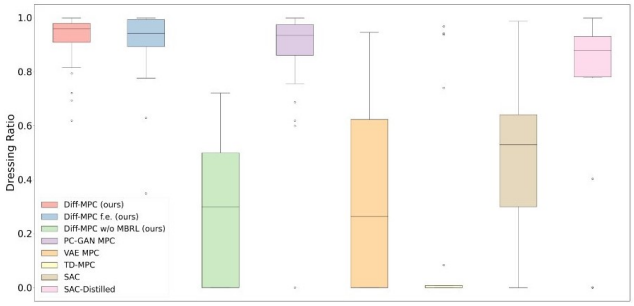


Fig. 3. Success rate measured as the achieved dressing ratio for the experiments performed in simulation. The boxplots represent the distribution of the success rates from 20 trials for each of the examined baselines. Our method achieves the highest performance compared to the baselines, while our method with a feature action encoding and the SAC policy trained with distillation follow closely.

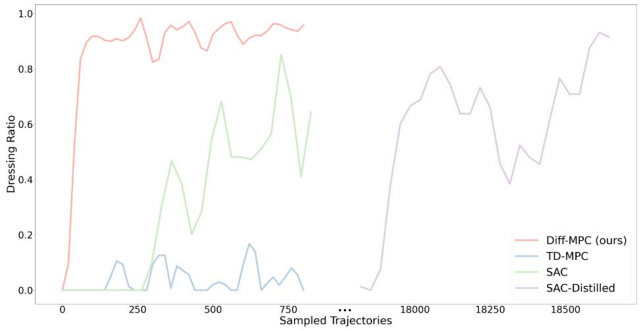


Fig. 4. Learning curves of our proposed method and baselines. The curves have been smoothed for the sake of visualization. For clarity, we note, that the SAC-distilled policy appears further in the plot, as the training of the 27 teacher policies used to guide the distilled policy has preceded. Our approach significantly outperforms the baselines in terms of sample efficiency, having achieved a 91.2% success rate within 100 sampled trajectories.

become inaccurate. From the two generative models used as baselines instead of the proposed Diffusion model, PC-GAN presented comparable performance, while the VAE achieved a relatively low average dressing ratio. TD-MPC achieved the lowest performance, failing to insert the opening around the hand in most cases. Finally, from the two model-free RL approaches, SAC could not generalize for the different configurations of the arms, as the garment was usually caught around the elbow of the arm. Distillation solved this issue and achieved a higher dressing ratio. However, as indicated by the learning curves, this method requires a high number of samples as several individual policies need to be trained for small ranges of the arm's configurations. Instead, Diff-MPC is very sample efficient, achieving a 91.2% success rate within 100 sampled trajectories.

B. Real-World

To investigate the transferability of our method from simulation to real-world dressing scenarios, as well as the generalization to novel poses and garments we deploy our approach in real-world dressing scenarios. In our experiments, we use a medical training manikin with movable joints. The manikin is equipped with a hand with an open palm, which adds to the difficulty of the dressing task, as the garments can often get caught in the fingers of the hand. We used 5 configurations of the arm from a wide range of poses. Specifically, two of the poses (P1,P2) belong in the distribution of poses used for training in the simulation, while three poses (P3-P5) are out of this distribution. We used 3 different garments for our experiments: a sleeveless jacket, a short-sleeved hospital gown, and a long-sleeved shirt. The poses and garments used are shown in Fig. 5. The manikin was fixated on a chair and the task was initialized by the robot grasping the garment around the opening's area. We initialized the robot-end effectors in a random position in front of the manikin's arm.

We compared our approach against all the baselines used for the simulation experiments. We also compare against a baseline (**Diff-MPC w/o DR**), in which we do not use domain randomization, to investigate the importance of this component for the sim-to-real transfer. We used the same proposed controller with the secondary motion for collision avoidance term to secure a fair comparison. We performed 3 trials for each combination of poses and garments, resulting in 45 trials per baseline. To measure the performance of the baselines and their sim-to-real transferability we measured the success rate, as well as the failure types, as shown in Fig. 7. Moreover, to investigate the generalization over different poses we measured the achieved dressing ratio per pose for the five most successful baselines, as shown in Fig. 6.

From the results shown in these two figures, the following conclusions can be drawn. Our method could generalize well to real-world scenarios, as the results in the real world follow the results of the experiments in simulation, indicating that the proposed diffusion dynamics can adapt to the sim-to-real gap of the garment's dynamics and the difference of the observations' point clouds. Moreover, our approach could generalize to the novel out-of-distribution poses used for evaluation. As for the baselines, the two examined action encoding methods of the proposed Diff-MPC achieved similar performance. The MPC without iterative learning behaved as in the simulation, by succeeding in the early stage of the task but failing to fully dress the arm. The PC-GAN dynamics, although achieving high performance in simulation, presented a low performance in the real-world tasks, while the VAE dynamics presented a similarly low performance as in the simulation evaluation. We note however, that a limitation of using a diffusion model is the larger inference time, measured at an average of 0.3554s on a Nvidia Geforce RTX 3090 GPU, compared to smaller inference times of PC-GAN (0.0008s) and VAE (0.0007s). A similar performance was observed when training the proposed controller without domain randomization. Therefore, regarding the proposed controller, we can conclude that using a diffusion model as dynamics had a great impact on the real-world performance and the sim-to-real adaptation, compared to the other examined generative models, using MBRL assisted with achieving high dressing ratios, while domain randomization did not have a great impact.

TD-MPC presented similarly low performance with the simulation and the same conclusions can be drawn. For the

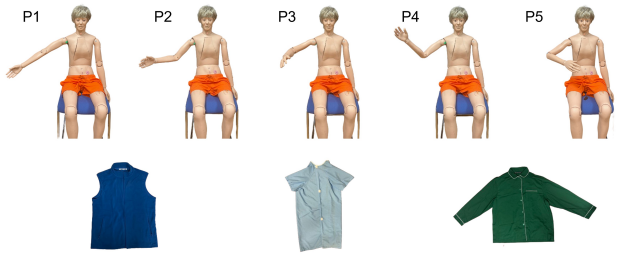


Fig. 5. Arm configurations and garments used during the real-world robot-assisted dressing experiments.

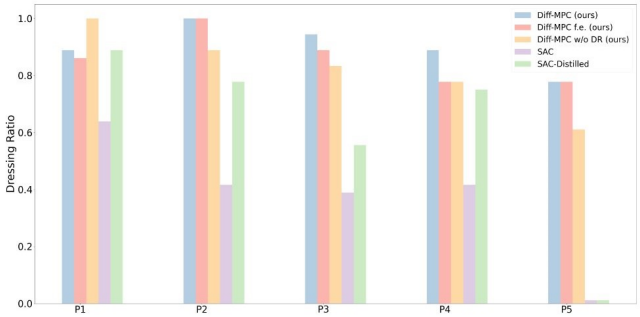


Fig. 6. Average dressing ratio achieved by the 5 most successful baselines over 9 trials per pose.

model-free RL baselines, we observed a lower performance compared to the simulation, indicating a lower sim-to-real adaptation. The garments were often caught on the palm of the manikin, while SAC often did not manage to turn the garment around the elbow of the manikin, which was also observed in the simulation. Guiding with distillation achieved a higher performance. However, as shown in Fig. 6, the learned policy was susceptible to poses with a large inwards elbow angle (P3,P5). We speculate that this is caused because these poses are far from the distribution of poses used for training. The fact that Diff-MPC could adapt and solve these out-of-distribution real-world scenarios indicates better sim-to-real adaptation and generalization to novel states.

V. Conclusions

In this paper, we proposed a garment diffusion dynamics model for robot-assisted dressing. The diffusion model was used to handle partial observations and generate future point cloud predictions. The dynamics model was utilized in a model predictive controller to select optimal actions. During training, we used model-based reinforcement learning to iteratively train our model with on-policy samples. In the real world, we deployed the model predictive controller as a primary objective and we used the redundant degrees of freedom to avoid collisions of the robot with the body. The conducted experiments demonstrate that our approach achieves high performance in simulated and real-world dressing scenarios. The dynamics model can be transferred in a zero-shot manner to real-world dressing applications, adapt to the reality gap and generalize well to novel configurations of the arms. Finally, the experiments show that the proposed approach is very efficient in terms of training. In future work, we aim to extend

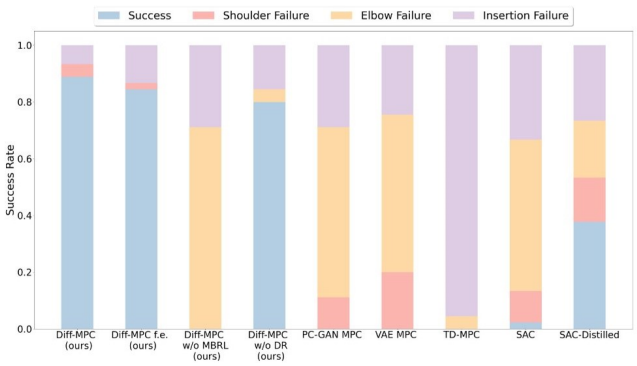


Fig. 7. Success rates and failure types of the proposed Diff-MPC and the baselines. The rates are measured over 45 trails with 5 different poses of the manikin and 3 garments. **Success**: The robot successfully dressed the garment up to the shoulder of the manikin. **Shoulder failure**: The robot successfully dressed the forearm and the elbow, but failed to dress the garment up to the shoulder of the manikin. **Elbow failure**: The robot failed to dress the garment beyond the elbow of the manikin. **Insertion failure**: The robot failed to insert the garment around the hand of the manikin.

our work for the case of non-static users, as well as study more complex dressing scenarios by incorporating grasping strategies. Future extensions also include investigating possible performance improvements by using the history of the point cloud observations.

References

[1] T. L. Mitzner, T. L. Chen, C. C. Kemp, and W. A. Rogers, "Identifying the potential for robotics to assist older adults in different living environments," *International journal of social robotics*, vol. 6, no. 2, pp. 213–227, 2014.

[2] F. Zhang, A. Cully, and Y. Demiris, "Personalized robot-assisted dressing using user modeling in latent spaces," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 3603–3610.

[3] Z. Erickson, H. M. Clever, G. Turk, C. K. Liu, and C. C. Kemp, "Deep haptic model predictive control for robot-assisted dressing," in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018, pp. 4437–4444.

[4] F. Zhang and Y. Demiris, "Learning grasping points for garment manipulation in robot-assisted dressing," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 9114–9120.

[5] Y. Wang, Z. Sun, Z. Erickson, and D. Held, "One policy to dress them all: Learning to dress people with diverse poses and garments." Robotics: Science and Systems, 2023.

[6] S. Kotsovolis and Y. Demiris, "Model predictive control with graph dynamics for garment opening insertion during robot-assisted dressing," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 883–890.

[7] Z. Sun, Y. Wang, D. Held, and Z. Erickson, "Force-constrained visual policy: Safe robot-assisted dressing via multi-modal sensing," *IEEE Robotics and Automation Letters*, 2024.

[8] A. Nagabandi, G. Kahn, R. S. Fearing, and S. Levine, "Neural network dynamics for model-based deep reinforcement learning with model-free fine-tuning," in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018, pp. 7559–7566.

[9] Z. Erickson, M. Collier, A. Kapusta, and C. C. Kemp, "Tracking human pose during robot-assisted dressing using single-axis capacitive proximity sensing," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2245–2252, 2018.

[10] F. Zhang, A. Cully, and Y. Demiris, "Probabilistic real-time user posture tracking for personalized robot-assisted dressing," *IEEE Transactions on Robotics*, vol. 35, no. 4, pp. 873–888, 2019.

[11] G. Chance, A. Jevtić, P. Caleb-Solly, G. Alenya, C. Torras, and S. Dogramadzi, ""elbows out"—predictive tracking of partially occluded pose for robot-assisted dressing," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3598–3605, 2018.

[12] Y. Gao, H. J. Chang, and Y. Demiris, "Iterative path optimisation for personalised dressing assistance using vision and force information," in *2016 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2016, pp. 4398–4403.

[13] A. Kapusta, Z. Erickson, H. M. Clever, W. Yu, C. K. Liu, G. Turk, and C. C. Kemp, "Personalized collaborative plans for robot-assisted dressing via optimization and simulation," *Autonomous Robots*, vol. 43, pp. 2183–2207, 2019.

[14] G. Canal, G. Alenyà, and C. Torras, "Adapting robot task planning to user preferences: an assistive shoe dressing example," *Autonomous Robots*, vol. 43, no. 6, pp. 1343–1356, 2019.

[15] F. Zhang and Y. Demiris, "Visual-tactile learning of garment unfolding for robot-assisted dressing," *IEEE Robotics and Automation Letters*, 2023.

[16] J. Qie, Y. Gao, R. Feng, X. Wang, J. Yang, E. Dasgupta, H. J. Chang, and Y. Chang, "Cross-domain representation learning for clothes unfolding in robot-assisted dressing," in *European Conference on Computer Vision*. Springer, 2022, pp. 658–671.

[17] F. Zhang and Y. Demiris, "Learning garment manipulation policies toward robot-assisted dressing," *Science robotics*, vol. 7, no. 65, p. eabm6010, 2022.

[18] T. Tamei, T. Matsubara, A. Rai, and T. Shibata, "Reinforcement learning of clothing assistance with a dual-arm robot," in *2011 11th IEEE-RAS International Conference on Humanoid Robots*. IEEE, 2011, pp. 733–738.

[19] T. Matsubara, D. Shinohara, and M. Kidode, "Reinforcement learning of a motor skill for wearing a t-shirt using topology coordinates," *Advanced Robotics*, vol. 27, no. 7, pp. 513–524, 2013.

[20] R. P. Joshi, N. Koganti, and T. Shibata, "A framework for robotic clothing assistance by imitation learning," *Advanced Robotics*, vol. 33, no. 22, pp. 1156–1174, 2019.

[21] S. Kotsovolis and Y. Demiris, "Bi-manual manipulation of multi-component garments towards robot-assisted dressing," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 9865–9871.

[22] Y. Wang, D. Held, and Z. Erickson, "Visual haptic reasoning: Estimating contact forces by observing deformable object interactions," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 11 426–11 433, 2022.

[23] K. J. Young, A. Ramesh, L. Kirsch, and J. Schmidhuber, "The benefits of model-based generalization in reinforcement learning," in *International Conference on Machine Learning*. PMLR, 2023, pp. 40 254–40 276.

[24] A. Brunnbauer, L. Berducci, A. Brandstätter, M. Lechner, R. Hasani, D. Rus, and R. Grosu, "Model-based versus model-free deep reinforcement learning for autonomous racing cars," *arXiv preprint arXiv:2103.04909*, 2021.

[25] R. Jangir, G. Alenya, and C. Torras, "Dynamic cloth manipulation with deep reinforcement learning," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 4630–4636.

[26] H. Ha and S. Song, "Flingbot: The unreasonable effectiveness of dynamic manipulation for cloth unfolding," in *Conference on Robot Learning*. PMLR, 2022, pp. 24–33.

[27] Y. Wu, W. Yan, T. Kurutach, L. Pinto, and P. Abbeel, "Learning to manipulate deformable objects without demonstrations," *arXiv preprint arXiv:1910.13439*, 2019.

[28] A. Longhini, M. Moletta, A. Reichlin, M. C. Welle, D. Held, Z. Erickson, and D. Kragic, "Edo-net: Learning elastic properties of deformable objects from graph dynamics," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 3875–3881.

[29] Y. Huang, C. Xia, X. Wang, and B. Liang, "Learning graph dynamics with external contact for deformable linear objects shape control," *IEEE Robotics and Automation Letters*, vol. 8, no. 6, pp. 3892–3899, 2023.

[30] R. Hoque, K. Shivakumar, S. Aeron, G. Deza, A. Ganapathi, A. Wong, J. Lee, A. Zeng, V. Vanhoucke, and K. Goldberg, "Learning to fold real garments with one arm: A case study in cloud-based robotics research," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 251–257.

[31] X. Lin, Y. Wang, Z. Huang, and D. Held, "Learning visible connectivity dynamics for cloth smoothing," in *Conference on Robot Learning*. PMLR, 2022, pp. 256–266.

[32] J. Urain, N. Funk, J. Peters, and G. Chalvatzaki, "Se (3)-diffusionfields: Learning smooth cost functions for joint grasp and motion optimization through diffusion," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 5923–5930.

[33] J. Carvalho, A. T. Le, M. Baierl, D. Koert, and J. Peters, "Motion planning diffusion: Learning and planning of robot motions with diffusion models," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 1916–1923.

[34] C. Chi, S. Feng, Y. Du, Z. Xu, E. Cousineau, B. Burchfiel, and S. Song, "Diffusion policy: Visuomotor policy learning via action diffusion," *arXiv preprint arXiv:2303.04137*, 2023.

[35] Z. Ding, A. Zhang, Y. Tian, and Q. Zheng, "Diffusion world model," *arXiv preprint arXiv:2402.03570*, 2024.

[36] M. Rigter, J. Yamada, and I. Posner, "World models via policy-guided trajectory diffusion," *arXiv preprint arXiv:2312.08533*, 2023.

[37] M. Janner, Y. Du, J. Tenenbaum, and S. Levine, "Planning with diffusion for flexible behavior synthesis," in *International Conference on Machine Learning*. PMLR, 2022, pp. 9902–9915.

[38] G. Zhou, S. Swaminathan, R. V. Raju, J. S. Guntupalli, W. Lehrach, J. Ortiz, A. Dedieu, M. Lázaro-Gredilla, and K. Murphy, "Diffusion model predictive control," *arXiv preprint arXiv:2410.05364*, 2024.

[39] P. M. Scheikl, N. Schreiber, C. Haas, N. Freymuth, G. Neumann, R. Lioutikov, and F. Mathis-Ullrich, "Movement primitive diffusion: Learning gentle robotic manipulation of deformable objects," *IEEE Robotics and Automation Letters*, 2024.

[40] A. Bartsch, A. Car, C. Avra, and A. B. Farimani, "Sculptdiff: Learning robotic clay sculpting from humans with goal conditioned diffusion policy," *arXiv preprint arXiv:2403.10401*, 2024.

[41] J. Yamada, S. Zhong, J. Collins, and I. Posner, "D-cubed: Latent diffusion trajectory optimisation for dexterous deformable manipulation," *arXiv preprint arXiv:2403.12861*, 2024.

[42] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 10 684–10 695.

[43] S. Luo and W. Hu, "Diffusion probabilistic models for 3d point cloud generation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 2837–2845.

[44] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 652–660.

[45] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, "Deep unsupervised learning using nonequilibrium thermodynamics," in *International conference on machine learning*. PMLR, 2015, pp. 2256–2265.

[46] G. Pavlakos, V. Choutas, N. Ghorbani, T. Bolkart, A. A. Osman, D. Tzionas, and M. J. Black, "Expressive body capture: 3d hands, face, and body from a single image," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 10 975–10 985.

[47] Z. Erickson, V. Gangaram, A. Kapusta, C. K. Liu, and C. C. Kemp, "Assistive gym: A physics simulation framework for assistive robotics," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 10 169–10 176.

[48] L. Žlajpah and T. Petrič, "Obstacle avoidance for redundant manipulators as control problem," *Serial and parallel robot manipulators-kinematics, dynamics, control and optimization*, pp. 203–230, 2012.

[49] H. Zhang, Y. Tian, Y. Zhang, M. Li, L. An, Z. Sun, and Y. Liu, "Pymaf-x: Towards well-aligned full-body model regression from monocular images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 10, pp. 12 287–12 303, 2023.

[50] C. Lugaresi, J. Tang, H. Nash, C. McClanahan, E. Uboweja, M. Hays, F. Zhang, C.-L. Chang, M. G. Yong, J. Lee *et al.*, "Mediapipe: A framework for building perception pipelines," *arXiv preprint arXiv:1906.08172*, 2019.

[51] P. Achlioptas, O. Diamanti, I. Mitliagkas, and L. Guibas, "Learning representations and generative models for 3d point clouds," in *International conference on machine learning*. PMLR, 2018, pp. 40–49.

[52] N. A. Hansen, H. Su, and X. Wang, "Temporal difference learning for model predictive control," in *International Conference on Machine Learning*. PMLR, 2022, pp. 8387–8406.

[53] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *International conference on machine learning*. PMLR, 2018, pp. 1861–1870.

[54] E. Liang, R. Liaw, R. Nishihara, P. Moritz, R. Fox, J. Gonzalez, K. Goldberg, and I. Stoica, "Ray rllib: A composable and scalable reinforcement learning library," *arXiv preprint arXiv:1712.09381*, vol. 85, p. 245, 2017.